A Dynamically Adaptable Image Processing Application Trading Off Between High Performance, Consumption and Dependability in Real Time

Juan Valverde, Alfonso Rodríguez, Javier Mora, César Castañares, Jorge Portilla, Eduardo de la Torre and Teresa Riesgo
Centro de Electrónica Industrial
Universidad Politécnica de Madrid
Madrid, Spain
email: {juan.valverde, alfonso.rodriguezm, javier.morad, jorge.portilla, eduardo.delatorre, teresa.riesgo}@upm.es

***Abstract*— As embedded systems evolve, problems inherent to technology become important limitations. In less than ten years, chips will exceed the maximum allowed power consumption affecting performance, since, even though the resources available per chip are increasing, frequency of operation has stalled. Besides, as the level of integration is increased, it is difficult to keep defect density under control, so new fault tolerant techniques are required. In this demo work, a new dynamically adaptable virtual architecture (ARTICo^3) to allow dynamic and context-aware use of resources is implemented in a high performance Wireless Sensor node (HiReCookie) to perform an image processing application.**

Keywords—Dynamic and Partial Reconfiguration, FPGAs, Wireless Sensor Networks, Parallel processing, Dependability, Image Processing.

I. Introduction

In the last few years, Cyber-Physical Systems (CPSs) have become very complex while they are now applied to a wide variety of environments [1]. Limited power consumption, fault tolerance and self-healing capabilities together with the need of working under changeable environments, impose the use of dynamic management of resources to ensure the proper delivery of service. Concepts such as urgency, confidentiality, fault tolerance or priority are widely known, but they are normally addressed as isolate objectives. The system in which this demo is implemented is capable of adapting its resources in real time depending on external and internal conditions in order to find a trade-off solution ranging among high performance, dependability and low power. This adaptability is based on Dynamic and Partial Reconfiguration (DPR) to replicate and/or change hardware accelerators.

The replication of modules, together with the adequate and efficient provision of data between accelerators and memories, can serve all these purposes. Actually, Double Module Redundancy (DMR), Triple Module Redundancy (TMR) and Side-Channel Attack (SCA) prevention techniques rely on hardware replication. Acceleration with parallel execution of threads is a well-studied model of computation, and there are tools and models, like CUDA, which may be used as a common specification entry for GPU platforms as well as for this type of HW accelerated architectures.

II. Hardware Platform & Virtual Architecture

The HiReCookie node [2] is normally composed of four stacked layers connected through a common vertical bus as shown in Fig.1. The layers are divided according to different functionalities so that they can be exchanged if necessary. The layers are communication (ZigBee, WI-FI, Ethernet or 802.15.4 versions), sensor layer (environmental parameters, video cameras, etc.), a power supply layer and the processing layer based on a Spartan 6 FPGA (LX150fgg484-2).

ARTICo^3 is a bus-based virtual architecture that can be

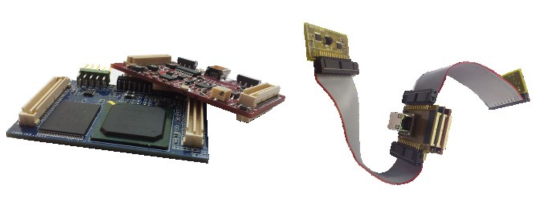

Fig.1. HiReCookie Node

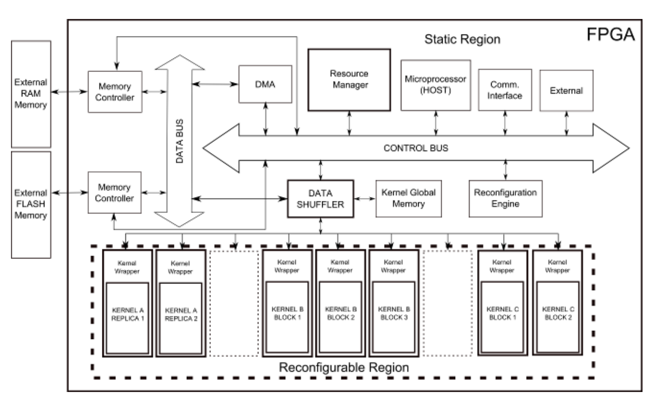

Fig.2. ARTICo^3 Architecture

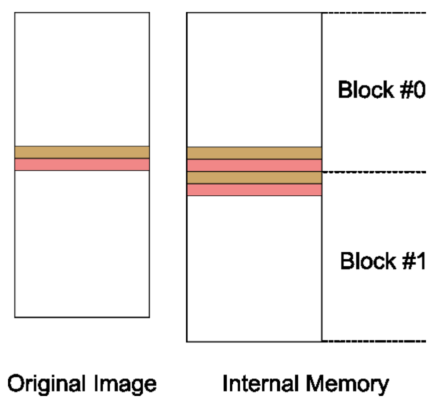

Fig.3. Image Acquisition and Internal Data Arrangement

used not only in the HiReCookie platform but in any other SRAM FPGA-based platform where a dynamic trade-off among consumption, dependability and high performance computation could be beneficial [3]. The general idea is taking advantage of DPR to be able to adjust resources in real time according to both external and internal conditions while the application code does not need to be aware of these changes. The architecture, which is shown in Fig.2, is divided into two different regions: static and dynamic. The static region includes those modules that are not reconfigured in real time while the dynamic one hosts different hardware accelerators that can be replicated or multiplexed in time depending on the working needs. In this demo, the hardware accelerators are image processing elements.

III. Demo Application

This demo presents an application for a dynamically adaptable median image filter. By using the HiReCookie-ARTICo^3 system, the application adapts its hardware resources according to external conditions to increase processing speed or recover from faults. In order to show different possibilities, the working point is changed remotely from a PC, increasing the level of noise introduced in the image, introducing faulty bitstreams or limiting the time available for processing. Images are taken by the video camera included in the HiReCookie node (Toshiba TCM8230MD) and filtered in real time. The results are then sent via Ethernet to the PC in order to see the whole process in a custom software application.

The image filter performs a convolution-based operation on the input image, using a moving window with a predefined size of 3x3 pixels. This is the basic functionality of each of the hardware accelerators that, in an analogy with the CUDA execution model, are called blocks. Each block is in charge of processing only a sub-region of the input image. Notice that blocks do not share data and, therefore, they must not have data dependencies. Taking advantage of a custom-made camera controller that has been specifically designed for the ARTICo^3 architecture, images are stored in the internal memory, replicating those rows located in the boundaries between sub-images without introducing additional latency. The resulting data arrangement requires additional memory space for these replicated rows. However, this procedure ensures that no data dependencies exist among processing elements (blocks). Fig.3 shows one possible data arrangement for the original input image located on the left when considering two sub-regions. The brown row is the last row of the first sub-image, and the red one is the first row of the second sub-image. The image acquisition system copies these two rows to generate the memory distribution located on the right. By replicating these two rows, no information is lost when the window moves through the sub-images, for the last output row of block #0 and the first output row of block #1 are not considered in the final solution (remember that in convolution-based processing applications, image borders are not well filtered and are usually not taken into account when comparing result qualities).

In this demo, the user can change at will the number of available slots in which the hardware blocks are placed or the number of blocks in which the processing has to be split in order to meet high performance requirements. In addition, the dependability requirements can also be changed, thus using more than one slot to provide the system with hardware redundancy and error detection and mitigation.

Acknowledgment

This work was supported by the Spanish Ministry of Economy and Competitiveness under the project DREAMS (Dynamically Reconfigurable Embedded Platforms for Networked Context-Aware Multimedia Systems) with number TEC2011-28666-C04-02.

References

[1] Edward A. Lee and Sanjit A. Seshia, Introduction to Embedded Systems, A Cyber-Physical Systems Approach, http://LeeSeshia.org, ISBN 978-0-557-70857-4, 2011

[2] Valverde, J.; Otero, A.; Lopez, M.; Portilla, J.; de la Torre, E.; Riesgo, T. Using SRAM Based FPGAs for Power-Aware High Performance Wireless Sensor Networks. *Sensors* 2012, *12*, 2667-2692

[3] Valverde. J; Rodriguez. A; Camarero. J; Otero. A;Portilla. J, de la Torre. E, Riesgo. T, "A Dynamically Adaptable Bus Architecture for Trading-Off Among Performance, Consumption and Dependability in Cyber-Physical Systems," *Field Programmable Logic and Applications (FPL), 2014 24th International Conference on.* (in press)